Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.777656

DETECTION AND CLASSIFICATION OF DDOS ATTACKS USING MACHINE LEARNING AND DEEP LEARNING

Govinda K
SCOPE, VIT, Vellore, India
kgovinda@vit.ac.in

Chintalapati Akhil
SCOPE, VIT, Vellore, India
chintalapati.akhil2020@vitstudent.ac.in

Abstract
DDoS attacks flood websites and online services with more traffic than servers and networks can handle. As technology has advanced and the Internet has become more widely used, such attacks have become more common and easier to carry out. A 2017 Cisco Visual Networking Index (VNI) report predicts almost an increase of about two times i.e., 14.5 million of the total number of DDoS attacks by the year 2022. Systems have been proposed for detecting DDoS attacks in the network in an efficient manner. Machine learning is once again being compared to other approaches like intrusion detection systems (IDS), which are commonly utilised for intruder detection and attack type classification. To identify and categorise assaults, the proposed system employs a combination of machine learning methods like xGBoos , KNN, Stochastic Gradient Descent, and Naive Bayes and CNN and deep learning approach. The results demonstrate that XGBoost has the best accuracy, whereas KNN has similar results. The study is strengthened using a hybridization process by employing a bidirectional LSTM (Long Short Term Memory) to obtain more accuracy than the prior method, because the existing method uses machine learning for detection.

Keywords: K-nearest, Naïve Bayes, Long Short Term Memory (LSTM), Neural Networks, Accuracy

Introduction

DDoS attacks combine several attacks using client/server technology to cause a denial of service. To build attack power, use a computer as an attack platform to conduct attacks on one or more targets. The conventional peer taupe attack mode has been modified with distributed denial of service attacks. There are no statistical norms for attack behaviour, because attackers employ common protocols and services. By protocol type and service alone, it is impossible to discern between attack and normal behaviour. It's not easy to find a distributed denial of service assault. The investigation is currently underway. The following tactics are used to defend against domestic and international DDoS attacks.

The number of source IP addresses, the target port, and magnetic flux density were utilised to describe the features of multi-sound attacks in the process of DDoS attacks. They really are Most attack flows can be distinguished by methods, but they employ fewer messages. information. The majority of them can just use the source IP address and the destination port. Because it does not identify a specific assault type, the detection rate is low. In forecasting, machine learning plays an essential role. DDoS attack detection using machine learning has

also proven fairly successful. progress. The majority of DDoS detection machine learning techniques are naive. Although the strategies in the preceding literature improve detection accuracy to some amount, they do not fully use the data stream context.

Related Works

People may obtain information from anywhere in the world using the internet technology, and it is also easier to distribute information. There are repercussions to the usage of the internet, in spite of its endless possibilities and advantages. Hackers and spammers(invaders) happen to be the largest source of annoyance and threat to the internet community. The threats of attacks and misuse of information in computer systems could be traced back by the actions of many malicious organizations. Because of the complexity of computer infrastructure, identifying the invader and intrusion is a challenge [1].

To make the suggested system run on resource-constrained devices, they limited its memory usage. They suggest two unique strategies, auxiliary shifting and early decision, to alleviate performance loss due to restrictions in processing power and memory. A large number of matching operations could be successful reduced on resource-constrained systems using the above two strategies. Studies reveal a maximum speedup of about 2.14 in scalable performance, with the help of an IoT item.[2].

A light weight methodology was suggested based on an investigation of node consumption in 6LowPAN in this research. This paper also looks at mesh-under and route-over routing schemes' 6LowPAN energy consuming models. Sensor nodes with unusual energy use were labelled as malicious attackers. The results of our simulations suggest that the proposed intrusion detection system can correctly and effectively detect malicious attempts [3].

An experimental example is shown to demonstrate the active learning method's considerable performance improvement over the standard supervised learning strategy. While ML techniques are being traditionally used for interference detection, human-in-the-loop ML is still in its early stages, which uses both machine and human intelligence to detect IoT intrusions [4].

Many simulations were automated for the sle purpose of comparing the performance and efficiencies of various models. Data mining tools WEKA and H2O were used to execute supervised and unsupervised learning models. The models (NSL-KDD) were trained and tested, with the help of interference dataset from an On-Line system. After that, the models are assessed for efficiency and accuracy. Supervised learning algorithms were able to outperform unsupervised learning methods, from the results. Because of their accuracy and training duration, Naive Bayes, Gradient Boost Machine, and Distribute Random Forest were proven as the best versions for DDoS detection. Both the Gradient Boost Machine and the Distribute Random Forest were examined further to see which parameters could produce improved accuracy [5].

This study suggests us two machine learning approaches known as decision trees and SVM, on a customized dataset, to discover and label attack signatures. Stock profiles and many DoS attack frameworks in WSNs are included in the collection. Decision trees had a greater true positive rate (99.86% vs 99.62%) and a smaller false positive rate (0.05% vs 0.09%) than SVM, as per experimental results [6].

We give a detailed literature evaluation of machine learning procedures that were utilised to look at present challenges in WSNs from 2002 to 2013. The pros and cons of every proposed algorithm are compared with the given task. A comparison chart is illustrated to assist WSN designers in designing machine learning answers that are appropriate for their particular application issues [7].

We employ XGBoost as a detection approach in SDN-based clouds in this research. The POX is employed as an SDN controller, SDN topologies designed with the help of Mininet, and an ideal DDoS assault environment simulated with Hyenae. The XGBoost classifier contrasts the flow packet data set collected by TcpDump to different classifiers, for DDoS detection. It is evident that our method is more precise, has a lower false positive rate, is faster, and expandable, from the detection findings [8].

Based on DDoS attack detection methodologies, we classify solutions and establish prerequisites for an effective solution. A novel architecture for detecting and mitigating DDoS works in a large-scale network is presented - a smart city that is constructed on SDN infrastructure based on our findings. Our suggested framework have the potential of detecting and mitigating DDoS attacks against specific applications. This paper makes two key contributions: analysing and explaining SDN-based DDoS attack detecting and mitigating systems, and its classification based on detection approaches; a proactive, SDN-based DDoS Defense Framework that exploits SDN's network security features (ProDefense) [9].

First, the current datasets are evaluated in depth and a new taxonomy for DDoS attacks is suggested in this work. Second, a new dataset, CICDDoS2019, is created, addressing all the current flaws. Finally, a new detection and family classifying strategies are offered which works on a collection of network flow attributes with the help of the generated dataset. Finally, the most relevant feature sets are listed for detecting various types of DDoS attacks, along with their respective weights [10].

Proposed Method

1st Method

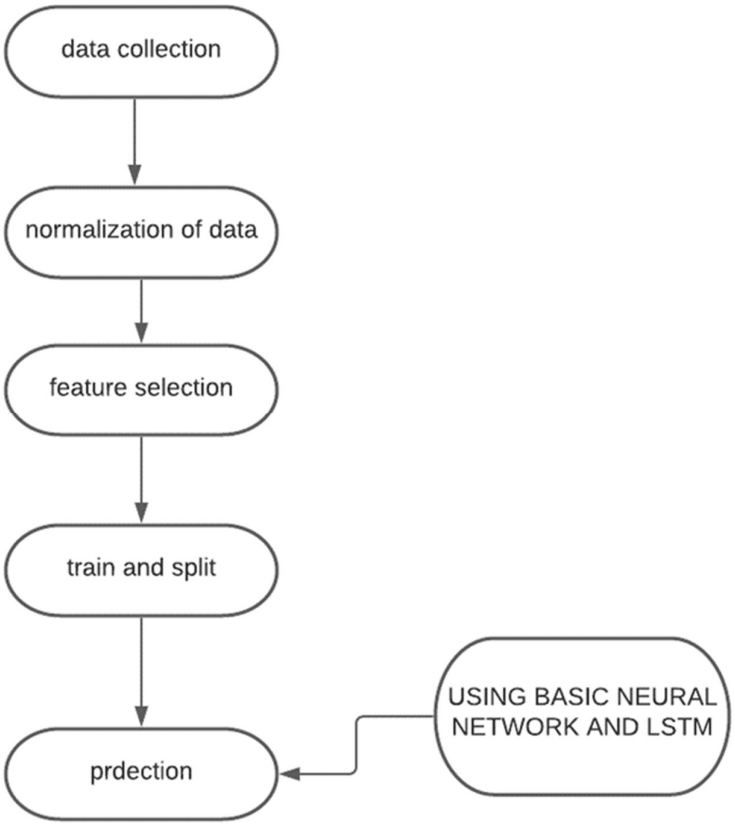

Fig1. Flowchart of Deep Learning technique

2nd Method

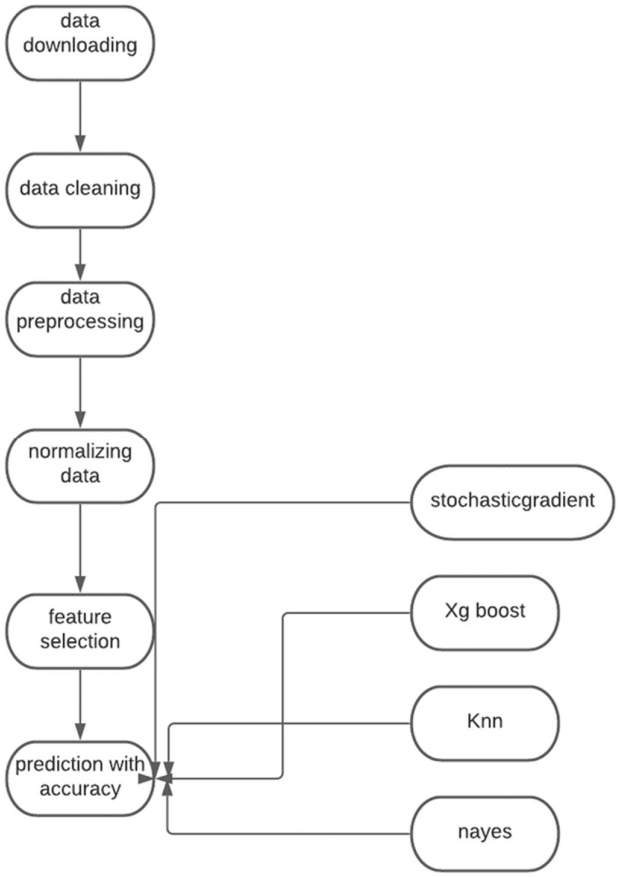

Fig2. Flowchart of the Machine learning algorithms

1st Method

BLSTM

1) The dataset is downloaded
2) Data is pre-processing is carried out
3) Data is selected with features with respect to BLSTM
4) Data is allotted for training and testing phase
5) Data is featured to plot the BRNN accuracy and loss plot
6) Data prediction confusion matrix is derived with graphs
7) Finally, data accuracy is predicted for the considered dataset KDD

Bidirectional LSTM

Bidirectional LSTMs train two LSTMs instead of just one on the input sequence in problems of input sequences, where timesteps for each are known i.e., the first LSTM on the original input sequence, and the second one on a reversed replica of it. This gives the network more context and helps it understand the problem quicker and more thoroughly.

Recurrent neural network (RNN)

The detection of network intrusion using a recurrent neural network is receiving more interest in the domain (RNN). The Recurrent Neural Network (RNN) is a sort of artificial neural

network that contains a linear graph between nodes with a temporal sequence. This class allowed a temporal exhibit to have dynamic behaviour.
The internal memory state RNNs can be utilised to process sequence inputs. RNNs are capable of memorising and perceiving by establishing a feedback loop that is related to the previous time step, which is distinct from feed-forward neural networks.
Some studies suggest a three-layer RNN with reduced size hubs that are only partially connected between layers.

2nd Method
Using large data set
1) The data set has been downloaded
2) Data has been preprocessed and cleaned
3) The dataset considered has been and again normalised
4) The normalised data set has taken for feature selection
5) After fracture selection data has been allotted into train and testing sets
6) Then classified using the machine learning algorithms
7) Finally got the best predicted algorithm for such large dataset!

K-Nearest Neighbour (KNN)
o The easiest Machine Learning algorithms, that extensively works on supervised learning approach.
o Assumes the similarity of upcoming case/data and existing cases and places the upcoming case in the category/class that is most comparable to the existing categories.
o Saves all existing data and classifies the fresh data points in accordance to their similarities over again.
o Used for both regression and classification. However, it is mostly used for the latter.
o Also called as a lazy learning algorithm as it does not learn from training set instantly: it saves the dataset and utilizes it to take actions on it during classification.
o Saves the dataset throughout the training period, and when new data is received, it classifies them into a category that is same as the new data. It is a non-parametric algorithm, that is it does not make assumptions about the data it utilizes.

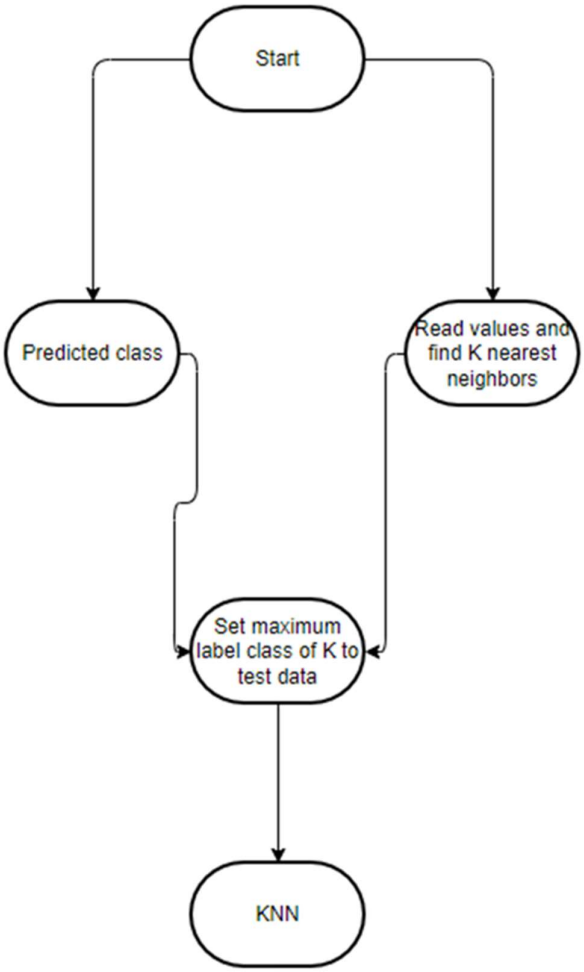

Fig3. Shows the flowchart of the KNN algorithm

Naïve Bayes Classifier Algorithm

o Supervised learning method that addresses classification problems based upon the Bayes theorem.

o Used in text classification with a high-dimension training dataset.

o Basic and productive classification method that helps in developing faster machine learning models competent of making quick predictions.

o Predicts based on an object's probability (probabilistic classifier).

XGBoost

o Machine learning method that has monopolized ML analysis for structured data.

o High-speed and high-performance execution of gradient boost decision trees.

Data Pre-processing

- Gather data
- Discover and assess data
- Cleanse and validate data
- Formatting and combining data
- Analysing the data

Results and Discussion

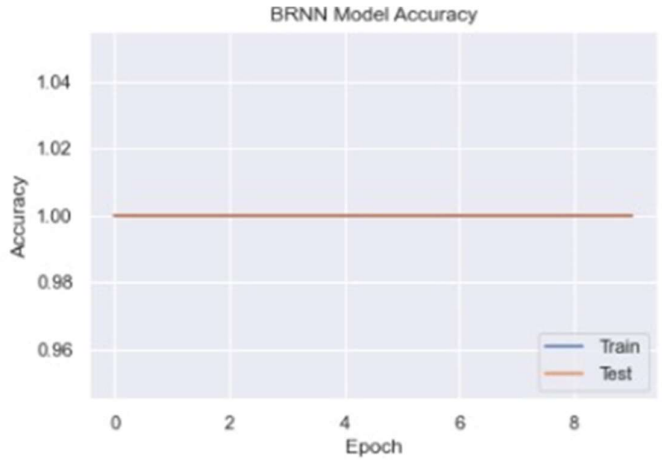

Fig4. Accuracy of the LSTM model.

To find the accuracy of the LSTM, the model accuracy is predicted by drawing the graph between number of epochs and accuracy of the model as shown in figure4. For the same model the loss decreases as the number epochs increased as shown in figure5.

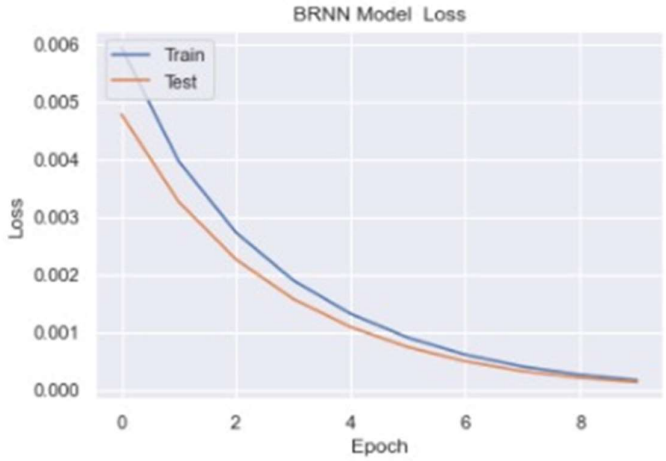

Fig5. BRNN Model Loss

```
scores = model.evaluate(X_test, Y_test, verbose=0)
print("%s: %.2f%%" % (model.metrics_names[1], scores[1]*100))
```

```
accuracy: 100.00%
```

Fig6. Code for accuracy of the LSTM model

In this method, we got the result by using Bidirectional LSTM. Only the accuracy has been considered and as the result shows it's 100%.

2nd Method:

```
Naive Bayes
Accuracy =  78.31658372130357
Confusion Matrix =
 [[14317     0     0     0     2     0     2     0]
 [    1 31901     2    10 17847     1     0     0]
 [    3     3 49751    55     7     3     0     0]
 [    0     0    52 48674    30     0  1517     2]
 [    8 20635    10    55 26188     1     7    14]
 [    6     0     0     1     1 49654     2     0]
 [    1     0     0  1076    14     1 49022    53]
 [    2     0     0    15    36     2   252   214]]
Recall = 0.8660170171777172
Classification Report =
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     14321
           1       0.61      0.64      0.62     49762
           2       1.00      1.00      1.00     49822
           3       0.98      0.97      0.97     50275
           4       0.59      0.56      0.58     46918
           5       1.00      1.00      1.00     49664
           6       0.96      0.98      0.97     50167
           7       0.76      0.41      0.53       521

    accuracy                           0.87    311450
   macro avg       0.86      0.82      0.83    311450
weighted avg       0.87      0.87      0.87    311450

F1 Score =  0.833979217373542
```

Fig7. Classification Report of Naïve Bayes algorithm (Accuracy, Recall and F1 Score are calculated)

```
K Nearest Neighbour Classifier
Accuracy =  87.00979290415796
Confusion Matrix =
 [[14317     0     0     0     2     0     2     0]
 [    1 36944     2    12 12802     1     0     0]
 [    3     4 49750    56     6     3     0     0]
 [    0     0    54 48840    26     0  1354     1]
 [    8 24436    11    67 22379     1     6    10]
 [    6     0     0     1     1 49654     2     0]
 [    1     0     0  1200    12     1 48915    38]
 [    2     0     0    15    39     2   270   193]]
Recall = 0.8700979290415797
Classification Report =
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     14321
           1       0.60      0.74      0.66     49762
           2       1.00      1.00      1.00     49822
           3       0.97      0.97      0.97     50275
           4       0.63      0.48      0.54     46918
           5       1.00      1.00      1.00     49664
           6       0.97      0.98      0.97     50167
           7       0.80      0.37      0.51       521

    accuracy                           0.87    311450
   macro avg       0.87      0.82      0.83    311450
weighted avg       0.87      0.87      0.87    311450

F1 Score =  0.8320559295494468
```

Fig8. Classification Report of KNN algorithm (Accuracy, Recall and F1 Score are calculated)

```
XGBoost Classifier
Accuracy =  89.29871033338685
Confusion Matrix =
 [[17136     0     0     0     0     0     0     0]
 [    1 56298     6     0  3467     0     0     1]
 [    0     0 59645    42    17     5     0     0]
 [    2     1    24 60168    22     0   199     0]
 [   11 34241     8    68 21904     1     0    16]
 [    1     0     0     2     1 59765     2     0]
 [    1     0     0  1621    11     0 58430    19]
 [    1     0     0    20    55     0   129   399]]
Recall = 0.8929871033338684
Classification Report =
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     17136
           1       0.62      0.94      0.75     59773
           2       1.00      1.00      1.00     59709
           3       0.97      1.00      0.98     60416
           4       0.86      0.39      0.54     56249
           5       1.00      1.00      1.00     59771
           6       0.99      0.97      0.98     60082
           7       0.92      0.66      0.77       604

    accuracy                           0.89    373740
   macro avg       0.92      0.87      0.88    373740
weighted avg       0.91      0.89      0.88    373740

F1 Score =  0.8773342009066785
```

Fig9. Classification Report of XGBoost algorithm (Accuracy, Recall and F1 Score are calculated)

In this method, multiple scores are considered such as accuracy, recall, f-measure. As mentioned above, we have used three algorithms namely, Naïve Bayes, KNN and XGBoost classifier.

In the 1st method, we have increased the accuracy by using BLSTM and its speed is high and also the efficiency is better as number of epochs are increased and compared to the existing methods. In the 2nd method, the accuracy can be obtained fast if we have a good processor and graphics user card in our systems. XGBoost has obtained more accuracy for large dataset as shown in table1.

Table1. Performance of models

Algorithms	Accuracy	Precision	Recall	F1-Score
RNN	100			
Naïve Bayes	78.316	0.76	0.86	0.833
KNN	87.009	0.80	0.87	0.832
XGBoost	89.298	0.92	0.89	0.877

Finally, the accuracy rate obtained has been increased a lot and also the time taken will be depending on the dataset used.

Conclusion

As we have analysed that data of DDOS is large in further, one can develop a software or apps on this and make available for all to use this, in the field of hybridizing the machine learning algorithms by using the Particle Swarm Optimisation(PSO) and Genetic Algorithm feature selections, as the result shows more accuracy, in case of deep learning, we can further more go and research into it and work the new RNN with LSTM and CNN based on the Artificial Intelligence for giving more advanced and highly efficient with high accuracy prediction of the attacks, which can further expand its wings into the Internet of things as well.

References

1. Suganth S, Usha D (2018) A survey of intrusion detection system in Iot devices. Int J Adv Res 6:23–30. https://doi.org/10.21474/ijar01/7183
2. Oh D, Kim D, Ro WW (2014) A malicious pattern detection engine for embedded security systems in the Internet of Things. Sensors 14(12):24188–24211. https://www.mdpi.com/1424-8220/14/12/24188
3. Lee T-H, Wen C-H, Chang L-H, Chiang H-S, Hsieh M-C (2014) A lightweight intrusion detection scheme based on energy consumption analysis in 6LowPAN. In: Huang Y-M, Chao H-C, Deng D-J, Park JJJH (eds) Advanced technologies, embedded and multimedia for human-centric computing, vol 260. Lecture Notes in Electrical Engineering. Springer, Netherlands, pp 1205–1213 https://link.springer.com/chapter/10.1007/978-94-007-7262-5_137
4. Yang K, Ren J, Zhu Y, Zhang W (2018) Active learning for wireless IoT intrusion detection. IEEE Wirel Commun 25:19–25. https://doi.org/10.1109/MWC.2017.1800079
5. Hoon K, Yeo KC, Azam S, Shanmugam B, Boer F (2018) Critical review of machine learning approaches to apply big data analytics in DDoS forensics. https://doi.org/10.1109/iccci.2018.8441286
6. Al-issa AI, Al-Akhras M, ALsahli MS, Alawairdhi M (2019) Using machine learning to detect DoS attacks in wireless sensor networks. In: 2019 IEEE jordan international joint conference on electrical engineering and information technology (JEEIT), Amman, Jordan, pp 107–112 https://ieeexplore.ieee.org/abstract/document/8717400
7. Alsheikh MA, Lin S, Niyato D, Tan H (2014) Machine learning in wireless sensor networks: algorithms, strategies, and applications. In: IEEE communications surveys and tutorials, vol 16, no 4, pp 1996–2018. Fourthquarter
8. Chen Z, Jiang F, Cheng Y, Gu X, Liu W, Peng J (2018) XGBoost classifier for DDoS attack detection and analysis in SDN-based cloud, pp 251–256. https://doi.org/10.1109/bigcomp.2018.00044
9. Bawany N, Shamsi J, Salah K (2017) DDoS attack detection and mitigation using SDN: methods, practices, and solutions. Arab J Sci Eng 42. https://doi.org/10.1007/s13369-017-2414-5
10. Sharafaldin I, Lashkari AH, Hakak S, Ghorbani AA (2019) Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy. In: IEEE 53rd international carnahan conference on security technology, Chennai, India https://ieeexplore.ieee.org/abstract/document/8888419

11. Graham JW (2009) Missing data analysis: making it work in the real world. Annu Rev Psychol 60:549–576 https://www.annualreviews.org/doi/abs/10.1146/annurev.psych.58.110405.085530
12. Kang H (2013) The prevention and handling of the missing data. Korean J Anesthesiol 64(5):402–406. https://doi.org/10.4097/kjae.2013.64.5.402CrossRefGoogle Scholar
13. Zhang H, Wei H, Tang Y, Pu Q (2019) Research on classification of scientific and technological documents based on Naive Bayes. In: Proceedings of the 2019 11th international conference on machine learning and computing (ICMLC '19). Association for Computing Machinery, New York, NY, USA, pp 327–331
https://dl.acm.org/doi/abs/10.1145/3318299.3318330
14. Fouladi RF, Seifpoor T, Anarim E (2013) Frequency characteristics of DoS and DDoS attacks. In: Signal processing and communications applications conference (SIU), 2013 21st. IEEE, pp 1–4 https://ieeexplore.ieee.org/abstract/document/6531200
15. Zhijun W, Yue M, Li D, Xie K (2015) Sedp-based detection of low-rate dos attacks. Int J Commun Syst 28(11):1772–1788 https://onlinelibrary.wiley.com/doi/abs/10.1002/dac.2783
16. Kun W, Jiahai Y, Fengjuan C, Li C, Wang Z, Yin H (2014) Two-stage detection algorithm for ROQ attack based on localized periodicity analysis of traffic anomaly. In: 2014 23rd international conference on computer communication and networks (ICCCN). IEEE, pp 1–6 https://ieeexplore.ieee.org/abstract/document/6911829
17. Perakovic D, Periša M, Cvitić I, Husnjak S (2016) Artificial neuron network implementation in detection and classification of DDoS traffic. https://doi.org/10.1109/telfor.2016.7818791
18. Saied A, Overill RE, Radzik T (2014) Artificial neural networks in the detection of known and unknown DDoS attacks: proof-of concept. Commun Comput Inf Sci 430:300–320 https://link.springer.com/chapter/10.1007/978-3-319-07767-3_28
19. Kale M (2014), DDOS attack detection based on an ensemble of neural classifier. Int J Comput Sci Netw Secur 14(7):122–129 http://cloud.politala.ac.id/politala/1.%20Jurusan/Teknik%20Informatika/19.%20e-journal/Jurnal%20Internasional%20TI/IJCSNS/2014%20Vol.%2014%20No.%2007/20140721_DDOS%20Attack%20Detection%20Based%20on%20an%20Ensemble%20of%20Neural%20Classifier.pdf
20. Jiadong Ren (2019)Building an Effective Intrusion Detection System by Using Hybrid Data Optimization Based on Machine Learning Algorithms. https://www.hindawi.com/journals/scn/2019/7130868/